Pick a nonce and try a hash

SECOND DRAFT OF ARTICLE FOR THE LONDON REVIEW OF BOOKS

How to mine bitcoins

Donald MacKenzie

Every time she's successful, a bitcoin 'miner' creates for herself 12.5 new bitcoins, currently worth around $80,000. If she doesn't succeed, she can have another go in roughly ten minutes time – all day, every day. Unsurprisingly, therefore, there's lots of mining going on worldwide. You can try mining on your laptop, but don't be too hopeful. Nowadays, to have a serious chance of winning the prize you need a specialised computer system – ideally, hundreds or thousands of them. The world's largest 'mine', run by a subsidiary of the Chinese company Bitmain in the high desert of Inner Mongolia, has over 20,000 of these machines.

Bitcoin mining uses a lot of electricity. Each individual machine consumes a kilowatt or more, around the same as a domestic electric heater. Indeed, a big headache for miners is keeping a warehouse packed with thousands of these machines (which is what a 'mine' is) cool enough to stop them breaking down. A May 2018 paper in the energy research magazine *Joule* estimated that bitcoin mining globally is consuming at least 2.5 gigawatts, which is getting close to the entirety of Ireland's electricity consumption. If the estimate is right, each individual bitcoin transaction indirectly requires, on average, 2-300 kilowatt hours of electricity, which is equivalent to leaving a heater running full-blast for four days or more. Given that most bitcoin transactions are tiny by the standards of global finance, that's strikingly profligate – and if bitcoin's dollar price resumes the upward trajectory it had in 2017 (it has fallen sharply in recent months), then the increased value of the 12.5 bitcoin prize would lead even larger amounts of electricity to be devoted to mining.

Mining was the way in which bitcoin's original designer, Satoshi Nakamoto, sought to solve the basic problem of any electronic currency: making sure that a user doesn't spend the same unit of currency more than once. Since the vast majority of pounds, dollars or euros are also now electronic, the issue isn't unique to bitcoin. The way in which the problem is usually solved is by keeping a centralised record of transactions, with tight controls over who can amend or add to the record. That, for example, is how your bank does it.

Satoshi didn't want to do it that way. Even though, famously, we don't know Nakamoto's real identity, it's clear from the original 2008 paper proposing bitcoin, and from the emails in which Satoshi discussed it, that Nakamoto was familiar with – and may have been part of – a strand of thinking within computer science that combined technical sophistication with fears about the invasion of privacy and a libertarian distrust of centralised authority. Bitcoin isn't a company, and it isn't even an organisation in any full sense. It's a software system. Satoshi – who might, perhaps, be a group of people, not just one individual – seems to have done all the initial programming. The system was then refined by other programmers, many of whom worked on the same voluntary basis as, for example, those who contribute to and police Wikipedia. Those programmers seemed – and many of them still seem – strongly committed to Satoshi's vision. A decade on, the central features of the bitcoin system remain almost entirely unchanged.

Satoshi's initial proposal had the title: 'Bitcoin: A Peer-to-Peer Electronic Cash System'. 'Peer-to-peer' wasn't just a libertarian aspiration: it signalled a particular type of technical configuration. In conventional electronic banking, for example, your laptop or mobile phone acts as an electronic 'client' computer interacting with a central computer server operated by your bank. In a peer-to-peer network, in contrast, each user's machine can be both client and server. The attractiveness of a peer-to-peer configuration had been boosted by the failure, in 1998, of an earlier form of electronic money, eCash, developed by computer scientist David

Chaum. When Chaum's firm, which ran the system in a centralised fashion, went bankrupt, it took eCash down with it.

In bitcoin, there is no central computer, and therefore no single point of failure, but also no central record. How can a decentralised, peer-to-peer network of computers construct a single, agreed record of transactions? After all, there's no barrier to anyone joining bitcoin, and some who join can be expected to be thieves or fraudsters. Mining – Satoshi's conceptually brilliant solution to this difficult problem – was an adaptation of an earlier proposal by, among others, the British programmer Adam Back. The bitcoin software system offers the 'prizes' (the mining rewards) I've mentioned so as to give many different bitcoin users an incentive to have their computers continuously check the validity of bitcoin transactions, pack these into an evolving public record of every bitcoin transaction that has ever taken place, and check others' additions to the record. After an hour or so has passed, the record becomes close to impossible to alter. That record is bitcoin's famous blockchain.

With an agreed record – a single version of history – in place, checking the validity of a transaction is straightforward. When you join bitcoin, you use its software to generate for yourself an anonymous electronic address, along with what's called a 'private key', which is a long string of binary digits that your computer uses electronically to sign a transaction, and an associated 'public key' that others can use to check the validity of that electronic signature. (You can generate as many different anonymous addresses as you like, each with its own equally anonymous digital keys.) What's then needed for it to be possible validly to send a given amount of bitcoin from one of these addresses is simply that the blockchain contains an earlier transaction in which the address has received that amount, and no transaction via which it has already been spent. A bitcoin is thus not a discrete thing – not even an electronic thing – that you own. It's simply a chain of transactions, always leading

back either to the ‘genesis block’ of 50 bitcoins mined by Satoshi in January 2009 or to subsequent successful mining.

When a bitcoin user initiates a transaction, her computer system dispatches a message embodying the transaction to other computers whose users have also joined the bitcoin network. Those systems retransmit the message, and eventually it reaches all or nearly all of the network. (Because there’s no central server, there’s no way of broadcasting a message directly to the entire network.) A miner’s computers gather together these messages into a block of around 2,000 transactions, ‘hashing’ them as they go.

Hashing is what miners’ computers spend most of their time doing, and how they do it explains both bitcoin’s chief technical achievement – a near-immutable, fully consensual record without a central record-keeper – and its alarming electricity consumption. A hashing algorithm takes a message (or other text), scrambles it thoroughly, and condenses it into a relatively short fixed-length form that’s called the ‘digest’. The algorithm used by bitcoin is known as SHA-256, one of a family of ‘secure hash algorithms’ based on research conducted by the US National Security Agency. The ‘256’ refers simply to the number of binary digits in the digest.

You don’t have to be overly paranoid to pause for a few seconds when you learn that the lineage of the crucial technical component of bitcoin includes an intelligence agency renowned as the world’s premier code-breakers. As far as I can see, though, there are no real grounds for worrying that the NSA has built in a subtle flaw so that it can decrypt messages scrambled using SHA-256. The algorithm was made public by the US National Institute of Standards and Technology, and the steps in it are simple enough that a ‘back door’ of this kind would be hard to conceal. By the time that SHA-256 was first released, in 2001, the NSA seems to have realised that it would be foolish to insert a back door into cryptographic

techniques that were going to be used widely in the civilian world. Those techniques are utterly central to everyday electronic commerce and to the global financial system. If the bad guys were to discover the back door, chaos would ensue.

Let's look at an example of an SHA-256 hash, expressed not as a long string of binary digits but (in what has become the standard written form) as 64 characters, each either a decimal digit or one of the first six letters of the alphabet. Here is the hash – the 'digest' – of *The Waste Land*: b7529e2290b3f69ecee705055c19e5d6891a1409aa02f0f3e5545a625bcace66.[1] If hashing the canon appeals, you can do it yourself, using, for example, the SHA-256 hash function you can find at passwordsgenerator.net. You'll be struck by how rapidly the hash appears, which indicates that for a modern digital computer SHA-256 hashing is a very straightforward operation. Crucially, though, it isn't 'invertible': even with all the computer power in the world, it would take you aeons to work back to the original message from the digest produced by a well-designed hash function.

You can also discover another important property of hashing by making a tiny alteration in the text being hashed. Change a single letter in *The Waste Land*, for example altering 'Starnbergersee' to 'Stirnbergersee', and you'll find that the new hash is completely different. In this case, it becomes: aa652e3ba70b42d129330e8c692f3b4f3f4ea1ac925526569dfa8739b1c082a9. That extreme sensitivity to the tiniest details of the input makes hashing an excellent technique for building a permanent record of transactions. What miners hash isn't simply the current block of transactions; they also incorporate the hash of the previous block; which in its turn includes its predecessor's hash; and so on backwards in time all the way to Satoshi's genesis block. Suppose just one aspect of one transaction is altered (perhaps several years ago someone had

[1] John Lanchester hashed Joyce's *Ulysses* for his article on bitcoin in the *LRB* of 21 April 2016.

received one bitcoin and now alters that to ten bitcoins). It isn't just the hash of that old block that would change radically. The hash of every subsequent block would do so too, making evident the fact that the blockchain had been tampered with.

One could imagine bitcoin working perfectly well technically with just a single miner doing all this hashing. Because SHA-256 hashing is not very demanding computationally, that miner would need only a standard computer, and the electricity consumed would be minimal. But a single miner is an idea entirely alien to Satoshi's vision: it would be a form of centralisation. That miner would indeed enjoy great power, for example to exclude transactions from the blockchain, demand excessive payments for including them, or alter their details.

Hence the need for multiple miners, each acting as a check on the others. The software of the bitcoin system, from Satoshi's days onwards, turns mining into a competition by requiring the miner not just to hash a block of transactions – that, as I've said, is easy – but to produce a hash that is a binary number below a certain threshold size: in effect, a hash that begins with at least a certain minimum number of zeros. Originally, the requirement was for only a small number of zeros, but as more and more computer power gets devoted to mining, the bitcoin software automatically increases the difficulty of the computation by reducing the size of the threshold and thus requiring a greater number of zeros. Here, to pluck an example at random, is the successful hash of block 540062, mined at 4:36pm (UK time) on September 5: 000000000000000001bc052e0aded766c4c6d4ab07608530de4c19f004f1c75.

That's utterly terrifying. If you translate that hash back into a string of binary digits, it begins with 75 zeros. You have to try a gigantic number of hashes before you can expect to find one like that, which is why mining consumes so much electricity. What miners have to hash includes not just a bundle of transactions but also what cryptographers call a 'nonce', an

arbitrary number. (It's an old word, found for example in *Hamlet*; 'for the nonce' meant 'for the occasion'.) There's no known way of predicting in advance the results of SHA-256 hashing, so the only way to find a hash with the requisite number of initial zeros is randomly to pick a nonce and try a hash. If that fails to produce the desired result, and it almost always will, then there's nothing better than to try again with a different nonce. Since bitcoin nonces are numbers with 32 binary digits, and there are over 4 billion such numbers, there's a lot of nonces you can try. Nowadays, given the very demanding nature of the goal, it's usual to find that not a single one of these nonces will work. If that happens, a miner's computer then turns to what is in effect a second nonce. It's a data field in the special 'coinbase' transaction that a miner always adds to a block, a transaction that creates the 12.5 new bitcoins if the miner is successful. The computer changes that second nonce, and then once again starts trying every possible value of the first nonce, and so on until it finally finds a hash with at least the required minimum number of zeros – or, more likely, until somebody else's computer does, in which case all this work is wasted, in the sense that it produces no reward.

It's true, though, that the computational intensity of mining is part of what makes the blockchain so difficult to alter: if you want to alter one block, you have in effect to re-mine not just that block but every subsequent block as well. Furthermore, precisely because there's no known way of finding a hash with the requisite number of zeros that's better than picking nonces at random, mining is not just hard but also a lottery. The latter aspect actually fitted Satoshi's peer-to-peer vision well. Any bitcoin user could leave her computer humming away gently – it's easy enough to make the process of mining entirely automatic – and every so often she would discover that she had a winning ticket. Even if that didn't happen, her computer would usefully have joined in the process of checking that's necessary to ensure a single version of history.

The snake in Satoshi's Eden turned out to be one of SHA-256 hashing's most attractive features, its computational simplicity. Its core operations don't require moving data between a computer's microprocessor chip and the computer's main memory, and the arithmetic involved is simply a form of the addition of whole numbers, so there's no need to use the microprocessor's 'floating point unit', which performs arithmetic with numbers that aren't integers. It was therefore soon realised that hashing could be 'parallelised', as a computer scientist would put it. Instead of doing hashes one after the other on a standard computer, a miner can employ other forms of hardware that have less flexibility but on which one can try multiple different hashes simultaneously, each with a different nonce.

The first person who is recorded as taking this approach to mining is a Hungarian-American programmer called Laszlo Hanecz (Nathaniel Popper tells the story in his history of bitcoin, *Digital Gold*). In 2010, Hanecz started employing a graphics processing chip of the kind used in computer game consoles. Generating an ever-changing image also involves doing large numbers of simple operations as quickly as possible – just what's needed for bitcoin mining. With his graphics chip, Hanecz overpowered the original bitcoin miners, who were using standard computers, and soon he was winning a quite disproportionate number of newly created bitcoins. In a message quoted by Popper, Satoshi successfully pressed Laszlo to curb his high-powered mining: 'I don't mean to sound like a socialist … I don't care if wealth is concentrated, but for now, we get more growth [of bitcoin] by giving that money [rewards for successful mining] to 100% of the people than giving it to 20%'. In 2010, bitcoin still had little or no dollar value, so it probably didn't seem too a big a sacrifice for Laszlo to comply.

Graphics processing chips did not in fact completely end mining's hobbyist phase. The technically-savvy young men who seem to have predominated among bitcoin's early users were, most likely, also computer gamers who were familiar with graphics chips. Learning how to run a hashing algorithm on a graphics chip was not too difficult for them, and it was

straightforward, and not hugely expensive, to buy these chips over the Internet or in computer hardware stores. You still had a chance of succeeding by buying half-a-dozen graphics chips, along with a fan or two to keep your kit cool (graphics chips burn a lot of electricity), and rigging up a simple mining operation.

What finally turned mining from an amateur into a predominantly professional activity was the introduction, from 2013 onwards, of ASICs, or application-specific integrated circuits. These are chips in which the circuitry to perform a specific task is etched directly into the silicon in the process of the chip's fabrication. Because SHA-256 hashing is such a simple operation, it's possible (although far from cheap) to design and have someone manufacture a chip that has many separate processor circuits, each of which hashes independently of the others. Although there are other firms also in the business, that's what Bitmain does. The chips that power its Inner Mongolian mine are of its own design, and are manufactured by the Taiwanese Semiconductor Manufacturing Company, owner of the world's largest silicon-chip foundry. Each of Bitmain's Antminer S9 machines contains 189 of these ASICs; in its turn, each of those ASICs has over a hundred separate little SHA-256 processor units hardwired into the chip.

There is therefore no hope of your laptop successfully competing against an Antminer. The current top-of-the-line version, the water-cooled S9 Hydro, can perform 18 billion hashes per second, and Bitmain is selling Hydros for a surprisingly modest $780 each. (Before you start to buy, remember that Bitmain seems currently to be earning more money by selling Antminers than by itself mining with them. As the saying goes: in a gold rush, sell shovels.) Each S9 Hydro gobbles up 1.7 kilowatts of electricity – that's why the water cooling comes in handy. But the enormous rate at which it hashes means that it uses far less electricity per hash than a standard computer or even a graphics chip.

Why, then, isn't bitcoin's global electricity consumption falling? The cheaper and more efficient hashing becomes, the larger the amounts of it miners in the aggregate do in order to try to win the prize. In part, that's simply a matter of the economics of this kind of competition, but there's also a further twist. Satoshi didn't want the bitcoin system to operate too fast. The rationale seems to be that – with no centralised form of broadcasting – the messages containing transactions and successfully hashed blocks of transactions percolate only relatively slowly through a globally-distributed network of computers. If mining was too fast a process, different segments of the network might start to treat different blocks as the one most recently mined, and so get out of synch with each other. History – the blockchain – could thus fragment ('fork', as a miner would put it) into multiple competing versions.

The bitcoin system is therefore designed to ensure that it takes around ten minutes on average before any miner anywhere manages to discover a nonce, or pair of nonces, that generates a hash with enough zeros. That makes mining a treadmill. Suppose the computer power devoted to mining increases. Blocks will then start successfully being hashed in less than ten minutes. That's when the adjustments I've mentioned kick in: the bitcoin software system simply increases the difficulty of the problem by requiring more zeros. (These adjustments happen every 2,016 blocks, or roughly every fortnight.) Sometimes – typically when bitcoin's price has fallen sharply – many miners find that they can't pay their electricity bills and so stop mining. If aggregate computer power goes down, the average time taken to mine each block starts to creep up, and the bitcoin system makes the problem easier. Since bitcoin's 2009 launch, though, most adjustments have required more, not fewer, zeros. That's how we got to block 540062 and its 75 zeros – and it helps explain why use of a much more efficient technology has ended up consuming more electricity, not less.

In the early summer of 1381, much of England was convulsed by insurrections of the common people. The townspeople of St Albans stormed its imposing Benedictine monastery, whose abbot was their feudal overlord. They burned the rolls, the records of the manorial courts. Rather more surprisingly, they also set about smashing the monastery's stone floors. Fifty years previously, its then abbot had finally succeeded in prohibiting the townspeople from milling grain by hand, and, as Marc Bloch records in *Land and Work in Medieval Europe*, '[f]rom all over the town the millstones were brought into the monastery, and the monks paved their parlours with them, like so many trophies'.

The confiscation of the St Albans millstones was an act of what, elaborating a term coined by the sociologists of science John Law and Annemarie Mol, we might call 'material political economy'. The abbot re-ordered the material world in a way that was economically consequential and was also political, in at least a broad sense of the word. Throughout the European middle ages, feudal lords such as the abbot often sought to suppress handmilling and replace it with windmills or watermills, because they were easier to police. If peasants or townspeople could mill in private, it was harder for their lords to exact what they regarded as their dues. Nor was the preference of the St Albans townspeople for handmilling – despite the physical effort involved – at all unusual. Even as wind and water were joined by steam power, handmilling continued. As late as the end of the nineteenth century, Bloch notes, 'Prussian villagers were still grinding grain' on handmills, and – even though landowners no longer had the right to prohibit handmilling – they still 'felt obliged … to hide from strangers as they did so'.

The material political economy of the mining of cryptocurrencies is more esoteric than that of the milling of grain: it does not determine who eats and who does not. Nor does it resemble conventional democratic politics: you 'vote' by either downloading and using a proposed new version of a cryptocurrency's software system, or by not doing so, and the influence of your

vote depends on the computer power at your disposal. But material political economy is what it is. The closest equivalent of the defence of handgrinding is the effort to design currencies with hashing algorithms that are, in the terminology of the field, 'ASIC-resistant' – in other words, algorithms for which it is hard to design specialised chips that will perform substantially better than ordinary computers. (A typical way of doing it is to try to ensure that the algorithm's operations, unlike those of SHA-256, need to make heavy use of a computer's main memory.) For example, the design of bitcoin's main rival, ethereum, included an attempt to make it ASIC-resistant.

Reordering the material world is, however, not easy work. The defence of the egalitarian, hobbyist mining of ethereum, for instance, has been only partially successful. It turns out that it is possible after all to design an ASIC chip for ethereum mining, although such chips haven't yet swept the board as their bitcoin equivalents have done. Efforts to change bitcoin itself have to contend with a particularly strongly entrenched status quo. Bitcoin's software *looks* malleable. It is open-source: anyone can download it, and if you have the appropriate skills – it would help if you are an experienced C++ programmer – anyone can modify it. But, as I've said, modifying a cryptocurrency's software is of no avail unless other users – especially the crucial users, the miners – start employing the new version. Switching from bitcoin's SHA-256 to an ASIC-resistant hashing algorithm is, for example, therefore politically unthinkable, because it would immediately render all those tens of thousands of Antminers and similar machines near-worthless.

Nor is it easy to persuade the relatively small group of programmers who can exercise a de facto veto on changes to bitcoin's core software. These men (and, as far as I can tell, they *are* mainly men) are not simply committed a priori to the central features they have inherited from Satoshi: they also know that the latter's vision has a certain coherence. Change one major feature, and other aspects of Satoshi's system could start to unravel. For instance,

making the problems that miners' computers have to solve easier wouldn't necessarily reduce aggregate electricity consumption (the way to win would still be to deploy the largest possible number of the most sophisticated machines), and – as I've already suggested – it could threaten the blockchain's coherence.

When proposals to alter bitcoin are canvassed, a particular fear that sometimes lurks in the background is of a 'majority attack', in which a single miner or group of miners deploys more computer power than the aggregate of all other bitcoin miners – which has indeed happened briefly at various times in the past – and uses it to make money not just by earning mining's legitimate rewards but by manipulating the evolving record of transactions (which has happened to other cryptocurrencies but not, so far, to bitcoin). A successful majority attack is a catastrophic event: it destroys a cryptocurrency's foundation, the agreed record of past transactions. To stop a majority attack becoming attractive, the rewards of honest mining need to be kept high, and what you can earn by manipulation kept low. That, as the economist Eric Budish has shown, places real constraints on how bitcoin can safely evolve. Budish's analysis, furthermore, suggests an irony. The undermining by specialist ASIC chips of Satoshi's egalitarian ideal may actually be helping protect bitcoin from majority attack, because gaining a majority of computer power would involve heavy investment in hardware for SHA-256 hashing that would lose much of its value when the price of bitcoin collapsed in the wake of a majority attack. (You can, it's true, rent mining hardware, but whether you could rent enough to mount a majority attack is doubtful.)

Even what looks to an outsider to be a minor technical change to the bitcoin system can spark fierce controversy among its miners and its core programmers. The system's deliberately slow pace means that it cannot process more than around seven transactions a second globally, and in practice the rate can be as low as two or three per second. (If you visit blockexplorer.com you can pretty much see the world's bitcoin transactions as they happen,

which – if you think about it – really shouldn't be the case. If there was a similar way of viewing the world's Visa or Mastercard transactions, all you could see would be a blur.) Yet all the proposals so far to change the bitcoin system in order to increase its capacity have foundered, often in the midst of acrimony – even the apparently very modest proposal to increase the maximum size of block from 1 megabyte to 2 megabytes. (Among the grounds for opposition to the proposal was again the fear that bigger blocks would percolate more slowly through the bitcoin network, causing miners to generate competing versions of recent history – which would facilitate exactly the kind of manipulation of the blockchain that currently requires a hugely expensive majority attack.) Those who design and who mine cryptocurrencies are intelligent people. They realise that bitcoin's limited capacity is a major constraint, and they can also see that there's something not quite right about huge amounts of electricity (much of it, alas, still produced from coal) being devoted to the trial-and-error solution of hugely daunting but ultimately arbitrary mathematical problems. But, as in ordinary politics, recognising a problem is not the same as agreeing what to do about it.

The most widely-canvassed alternative to the form of mining used in bitcoin (which those involved call 'proof-of-work') is what's known as 'proof-of-stake': ethereum's developers, for example, have said they intend to shift to the latter. In proof-of-stake, a cryptocurrency's software system randomly chooses a user and offers that user's computer the opportunity to be the one that hashes the current block of transactions and earns the associated reward. Mining wouldn't then take the form of a race (as it does with bitcoin), and there would be no need for specialised hardware or to make the problem artificially hard so that the race isn't over too quickly. You do, though, still have to worry that the user who gets selected in proof-of-stake might try to manipulate the evolving blockchain. That's where 'stake' comes in: proposals include requiring the chosen miner to make a chunky security deposit, and/or choosing a form of lottery that's most likely to be won by a user who has heavy investments

in the currency and who is thus less likely to take actions that could be expected to cause the currency to lose value. There are, however, still some who doubt that measures such as this would be enough to keep proof-of-stake secure, and more than a few who think it is inherently plutocratic.

I've focused on the material political economy of bitcoin mining, but there are other aspects of bitcoin that are also political – again in a broad sense of the word. You might think, for example, that each bitcoin would be worth the same as every other bitcoin – that, after all, is how money is supposed to work. But the history of a particular bitcoin matters. A dollar bill can bear the trace of its history (cocaine, explosives …), but a bitcoin *is* its history: as I've said, it's simply a chain of transactions. Although the latter are anonymous, they are recorded, publicly and indelibly, in the blockchain.

Sometimes, the chain that constitutes a particular amount of bitcoin includes a bitcoin address that has been discovered to have been used in theft, money laundering, the sale of weapons or illicit drugs, and so on. Bitcoin traders refer to such bitcoins as 'tainted'. You can try to remove taint by using a 'tumbler' or 'mixing service', which receives coins from multiple addresses and jumbles them before returning them, but such a service can simply spread a diluted form of the taint rather than eliminating it. The fear of taint – of, for example, a lawsuit demanding the return of allegedly stolen coins – is a barrier to mainstream financial organisations such as institutional investors becoming involved in bitcoin. Currently, institutional investors are reported as paying a premium of around 20 percent to buy, direct from miners, the new coins that make up the prize for successful mining, because these coins are free of history and therefore of the risk of taint.

In November 2008, a participant in the cryptography email list to which Nakamoto sent his original bitcoin proposal objected: 'You will not find a solution to political problems in

cryptography'. Satoshi's reply was vanilla libertarianism: 'Yes, but we can … gain a new territory of freedom for several years. Governments are good at cutting off the heads of a centrally controlled networks [sic] like Napster, but pure P2P [peer-to-peer] networks like Gnutella and Tor seem to be holding their own.' Bitcoin has done a great deal better than just hold its own, but Satoshi's critic has turned out to be right. Politics saturates bitcoin and the numerous rival cryptocurrencies it has inspired, and whether and how their political problems can be solved remain open questions.